Topological SLAM Using a Graph-Matching Based Method on Omnidirectional Images

Anna Romero, Miguel Cazorla
Instituto Universitario Investigación en Informática
Universidad de Alicante
aromero@dccia.ua.es, miguel.cazorla@ua.es

Abstract

Image feature extraction and matching is useful in many areas of robotics such as object and scene recognition, autonomous navigation, SLAM and so on. This paper describes a new approach to the problem of matching features and its application in scene recognition and topological SLAM. For that purpose we propose a previous image segmentation into regions in order to group the extracted features in a graph so that each graph defines a single region of the image. The matching process will take into account the features and the structure (graph) using the GTM algorithm. Then, using this method of comparing images, we propose an algorithm to construct topological maps. During the experimentation phase we will test the robustness of the method and its ability constructing topological maps.

1 Introduction

Features extraction and matching is an important area in robotics since it allows, among other things, object and scene recognition and its application to object localization, autonomous navigation, obstacle avoidance, topological SLAM. The SLAM (Simultaneous Localization And Mapping) problem consists of estimating the position of the robot while building the environment map. The solution to this problem is not trivial, since the error in the position estimation affects the map and vice versa. In the literature, depending on the form to represent the environment in which the robot moves, we can talk of two types of SLAM: the Metric SLAM and the Topological SLAM. In the first, the position is determined by a continuous space, *i.e.* we know exactly what position has the robot on the map. It is easy to find solutions that include odometry, sonars and lasers ([1, 2]). There are less solutions using vision since the calculation of the exact position is more complicated. In the second type, the different points where you can find the robot are represented by a list of positions, *i.e.* the map is a discrete set of locations which defines a small region on the environment. In this case there are plenty of solutions that use images for the calculations. In [3] they use the images captured by the AIBO robot to learn the topological map. We also find solutions using omnidirectional images such as [4] and [5], [6] where topological map is constructed using an incremental algorithm.

For both object and scene recognition we need methods to extract features from images. Several solutions in the literature use different methods for extracting the features. In [8] uses an over-segmentation algorithm for split the image into small regions. In paper [9] combines the Harris corner detector with SIFT descriptor. Many solutions in the literature are based on the combination of a segmentation algorithm with a feature extractor ([8], [10], [11]).

Object recognition requires a manually selected database to describe the objects that the robot must recognize. In the case of scene recognition we could need a scene database as in [12]. It introduces the concept of "Visual Place Categorization" (VPC) which consists of identifying the semantic category of one place/room using visual information. However, there are situations requiring no previous database as it is constructed as the robot navigates through the environment ([10], [13]) such as in the SLAM problem.

Affine invariant feature detectors have shown to be very useful in several computer vision applications, like object recognition and categorization, wide baseline stereo and robot localization. These detection algorithms extract visual features from images that are invariant to image transformations like illumination change, rotation, scale and slight viewpoint change. High level vision tasks that rely on these visual features are more robust to these transformations and also to the presence of clutter and occlusions. A more detailed survey of the state of the art of visual feature detectors can be found in [14]. In this work, the authors assess the performance of different algorithms for the matching problem, being the Maximally Stable Extremal Regions algorithm (MSER) [15], the Harris affine and the Hessian affine [16] the best suited for that task. Several methods are based on a combination of feature detectors (regions, contours and/or invariant points) to improve the matching and taking advantage of the extracting methods used in addition to eliminating some of the problems of the individual methods. However, it has not proposed the creation of structures form the extracted features to check the overall consistency of the matchings but the features are matched one by one without taking into account any possible neighborhood relationships. Some of those methods apply a matching consistency, eliminating cross-matches, those matches that intersects with other ones. In the case of omnidirectional images that can not be done, due to the circular nature of the images.

In this paper we propose a method for matching features and an algorithm to construct topological maps using this comparison method. For the image comparison method we propose a image preprocessing in two steps: segmentation into regions (using JSEG) and invariant feature extraction (using MSER with SIFT descriptors). Each of the obtained regions in the first step will contain a list of invariant points inside its domain. For each region, our method will construct a graph with the invariant points. The feature matching are made by comparing the graph of each of the regions of the current image with the representative graphs of the previously captured images. This approach takes into account both the feature descriptors and the structure of those features within the region. We apply the image comparison method in our topological map algorithm in order to group images that are considered belong to the same area.

The rest of the paper is organized as follows: Section 2 describes the preprocessing done to the image (JSEG segmentation) and feature extraction (MSER). Section 3 explains the graph matching using the GTM algorithm. Then, in section 4 we describe the algorithm that constructs topological maps. In 5 we present the results obtained applying the combination of the image matching method and the topological mapping algorithm. Finally in 6 some conclusions are drawn.

2 Image Processing

MSER (Maximally Stable Extremal Regions) [15] is an affine invariant shape descriptor. The MSER algorithm detects regions that are darker or brighter than their surroundings and can be scale invariant. The algorithm uses the SIFT [7] descriptor (not the detector just the descriptor) to describe the detected regions. Due to the nature of the descriptors, it is possible to associate (match) an MSER region (feature) of an image with one that appears in another image. Despite the robustness of the method we can find many cases where the feature matching has not been successful (false positives or outliers). To eliminate these false positives and thus obtain a more reliable and robust results in identifying scenes seen before, we propose to use a structure (graph) with which to compare (and match) images. To detect the different regions (which eventually form the sub-graphs to compare) of the image we use the segmentation algorithm JSEG.

2.1 Segmentation

Feature detection and extraction methods detect features along the whole image. Our goal is to group features according to the image region to which they belong, so we need a segmentation algorithm to divide an image into regions. In our case we use the one proposed in [17] known as JSEG algorithm.

JSEG finds homogeneity of a particular color pattern and texture. It assumes that the image:

- It contains a set of regions with approximately the same color and texture.
- Color data in each region of the image can be represented with a small set of quantized colors.
- Colors between two neighboring regions are distinguishable from each other.

In order to obtain different regions, JSEG performs segmentation in two steps: color quantization and spacial segmentation. In the first step of the algorithm, image colors are coarsely quantized without degrading the image quality significantly. This will extract a few representative colors that can be used as classes which separate regions of the image. For each image pixel, the algorithm find its class and replace its value, building an image of labels (class-map).

In the second step, a spatial segmentation is performed directly on the class-map without taking into account the color similarity of the corresponding pixel. This transforms the output from the previous step in a J-Image ([17]). Once this image is calculated, the algorithm uses a region growing method for image segmentation. Initially, the JSEG considers the image as one region, performs an initial segmentation with the scale and repeat the same process with the new regions and the next scale. Once the seed-growing step ends, the regions that have been over-segmented are merged with a grouping method. Finally we get two images, one where each pixel has the value of the belonging region and one with the real image which have overlapped the edges of each region.

An advantage of separating the segmentation into two steps yields an increase in the processing speed of each of the steps, which together do not exceed the time required for processing the whole problem. Furthermore, the process is not supervised and therefore there are no need for experiments to calculate thresholds, since the algorithm automatically determines them.

2.2 Feature Detection and Extraction

Once the different regions of the image are determined, we proceed to extract image features. In our case we use the affine invariant shape descriptor MSER.

The algorithm by Matas *et al* [15] searches extremal regions, that is, regions in which all pixels are brighter or darker than all the pixels in their neighborhood. The image pixels are taken in intensity order, forming connected component regions that grow and merge, until all pixels have been selected. From all these connected components, or extremal regions, the algorithm selects those for which size remains constant during a given number of iterations. Finally, the selected Maximally Stable Extremal Regions, that can have any arbitrary shape, are transformed into ellipses.

For each image, the features of the entire image are acquired and stored to build a representative graph of the image. Furthermore, each feature is assigned to the region it belongs (by using the position of the feature at the image), obtaining a set of invariant points for every region calculated in the segmentation step. Points that belong to the same region are those used to construct the various sub-graphs that describe the image, *i.e.* each region has its own graph, built with all the features in its domain. Note that it is possible that some regions have not associated any feature or some points do not belong to any particular region, in this case the region (or points) is discarded because it does not contain any interesting data.

3 Matching with Graphs

The feature matching process could result in unwanted false positives. To eliminate these outliers we suggest the use of graphs as structure for matching. The use of graphs allows checking not only a single invariant point consistency, but also a set of points that somehow have a relationship with each other.

The selected method for graph matching is GTM [18] (Graph Transformation Matching). This algorithm needs as input a list of the position of the matched points (x_1, y_1) (x_2, y_2). This list is calculated as follows:

- A $KD - Tree$ is built (this tree structure, $KD{-}tree$, allows relatively quick insertions and searches for a k-dimensional space (128 dimensions in our case, the SIFT descriptor dimension)) with all points of the base image (all points that forms the representative graph).
- For each region of the current image (image sub-graphs):
  - For each point in the region, the closest one in the $KD - Tree$ is found. If its Euclidean distance is below a threshold, we have found a match.

Once this step is completed, we have a list of matched points that describe a common region between the two images. As this matching may result in many false positives we use the GTM algorithm to compare the structure of the region in both images to eliminate those false positives in the matching.

GTM is a point matching algorithm based on attributed graphs [18] that uses information from the local structure (graph) for the treatment of outliers. The constructed graph for comparison is called K-Nearest-Neighbours which is built by adding an edge to the adjacency matrix for the pair (i, j) if node j is one of the k nearest neighbors of node i and if the Euclidean distance between the two points is also less than the average distance of all points on the graph. If a node has not k edges, it is disconnected until we finish the graph construction.

Once the two graphs from the two images have been constructed, the algorithm eliminates iteratively the correspondences distorting neighborhood relations. To do this, what is considered an outlier is selected, the two nodes (invariant points) that form the match (false positive) are removed from their respective graphs and also the references to those nodes in the two adjacency matrices. Then, the two graphs are again recalculated. The process continues until the residual matrix (the difference between the adjacency matrices of two graphs) is zero. At this point it is considered that the algorithm has found a consensus graph. Once this consensus graph is acquired, the disconnected nodes are eliminated of the initial matching, obtaining a match where the false positives are removed.

4 Topological Mapping

The results of the previous section allow us to know if two images can be seen as part of the same region (they have been taken at nearby positions in the real world). Using this method for image comparison we have built an algorithm capable of creating topological maps from a sequence of images that form a path in the real world. Our algorithm does not require a database because it is created as a new image is captured.

The algorithm builds topological maps in the form of undirected graphs that can be used in applications of topological SLAM. The topological map consists of nodes representing a particular area of the environment and a adjacency matrix that shows the relationships between them. The nodes can be composed of any number of images, but always have a representative image. This image is one that has more regions in common with the rest of images belonging to the node. In order to calculate the node representative and its minimum matching percentage, we use the formulas:

$$R = \arg\max_{i \epsilon I}(\min_{j \epsilon I, i \neq j}(C(i,j)) \qquad (1)$$

$$N_R = \max_{i \epsilon I}(\min_{j \epsilon I, i \neq j}(C(i,j)) \qquad (2)$$

This formulas appeared in [5] and use the number of matched points in function $C(i,j)$. In order to use our previous method, we have modified this formula which is as follows:

$$C(i,j) = \frac{Number\ of\ matched\ points}{min(NP_i, NP_j)} \quad (3)$$

where NP_k is the number of points in image k. We select the image with less number of points since it will match at most this number of points that otherwise could not reach 100% in the equation.

The algorithm builds the topological map as follows:

1. When we get a new image, it checks if the image belongs to the region that defines the current node. For this, this new image is compared with the node representative and if the matching percentage passes a certain threshold, it is added to the node.

2. If the image does not exceed the threshold, it is compared with all the node representatives, to find the node whose percentage is higher. If the comparison with the node above passes the threshold, the image is added to the node and creates an edge between the previous node and the newly found node.

3. If no match is found we establish that we have seen a new node, so it creates it and add an edge between new node and the previous one.

4. In any case, if we add an image to an existing node, if $Th_{min} \leq C(i,j) \leq N_R$, the node representative is re-calculated.

5 Results

This section shows the results of applying the whole algorithm on the set of images described in [19] which are available for download from the website of the authors. The images are omnidirectional, with a resolution of 2048x618. The tests were conducted on the images for the first route, the first 3000 of the data-set. Omnidirectional images have a sort of special features not found in other images. When images covering an angle of 360° it is possible to find two images from the same scene containing objects in a different situation.

In our previous work [20] we test the reliability and robustness of graph-based matching algorithm and in [21] we compare the response of the algorithm using different feature detectors. For this article we used the MSER detector since it was the best the algorithm that obtains the best result during the experimentation.

In figure 1 we have the graph representing the topological map created by our algorithm. Due to the large number of images, they have been processed one of every 15. The circles in the image represent the positions of the node representative and the arcs are the relations between the nodes (the edges of the graph). The threshold for considering a pair of images belong to same region has been estimated empirically.

As we can see, even though the path is composed by several laps in the environment, the topological map has not allocated different nodes for each of the laps made, but to consider the loop-closure existing and combined the images of the same area but caught at different times in a single node. However, there are some situations where for the same area the algorithm has created multiple nodes. In some cases it is because we have not taken pictures in a row, but in other cases is due to changes in light and/or changes in focusing direction. In figure 1 we see that the area of trees labeled as sample 1 in the image above has created multiple nodes. Looking at the two nodes representatives (figure 2), the images do not have the same illumination, are taken in opposite directions and also in the second image the tree cover parts that could be useful for identification (occlusion of many interest points). One can think that omnidirectional images avoid the rotation problem, but our algorithm does not have into account the fact that left hand side of the image is continuous with the right hand side. We are planning to solve this in future work.

Something similar occurs at the tunnel area

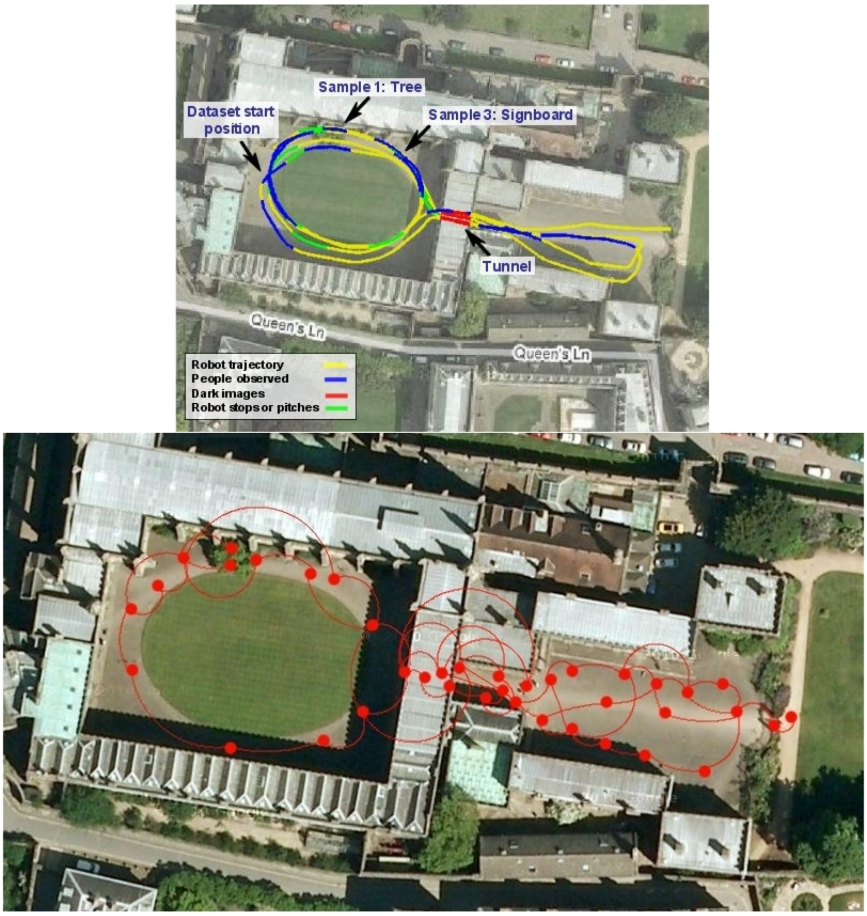

Figure 1: In the image above the route taken (from the authors page [19]). Below the graph topology generated by our algorithm.

but in this case, images inside the tunnel have very dark regions so there is no detection of so much feature points in order to match them.

6 Conclusions

In this paper we have presented a new method for creating topological maps. The method consists of the combination of image segmentation into regions and the extraction of feature detectors, and then the creation of a graph structure from those features. Thus, the matching of two images is taken into account both the extracted features and the structure (graph) formed by these features. Then, we construct the topological map using this comparison method, obtaining a non-directed graph that divides the environment in regions and indicates the relationships between different areas.

During the experimentation phase we have constructed a topological graph that describes the environment captured during a long path and with several loop-closures (several laps). As we have seen the environment is divided into several areas, most of them unique, that

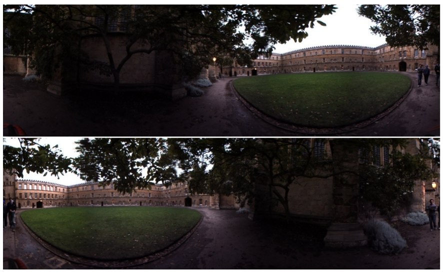

Figure 2: Above: image representative of node 2 (sample 1: tree). Below: image representative of node 28 (sample 1: tree).

is, are described by a single node. In cases where they have appeared more than one node we have seen changes due to illumination and/or rotation.

As future work, we plan to improve the algorithm in order to reduce the sensitiveness to changes in illumination and rotation. We also intend to make a more advanced study of the behavior of the algorithm using different features (SIFT, SURF, Harris-Affine, Hessian-Affine). We want also to extend our graphs in a circular way, in order to take into account the circular property of this kind of images.

7 Acknowledgements

This work has been supported by grant DPI2009-07144 from Ministerio de Ciencia e Innovacion of the Spanish Government.

References

[1] M. Montemerlo and S. Thrun, ”Simultaneous localization and mapping with unknown data association using FastSLAM“, in *Proc. of Intl. Conf. on Robotics and Automation*, vol. 2, pp. 1985-1991, Taiwan, 2003

[2] A. Diosi and L. Kleeman, ”Advanced Sonar and Laser Range Finder Fusion for Simultaneous Localization and Mapping“, in *Proc. of Intl. Conf. on Intelligent Robots and Systems*, vol. 2, pp. 1854-1859, Japan, 2004.

[3] Elvina Motard, Bogdan Raducanu, Viviane Cadenat and Jordi Vitrià, ”Incremental On-Line Topological Map Learning for A Visual Homing Application“, In *International Conference on Robotics and Automation*, pp. 2049-2054, IEEE, 2007.

[4] T. Goedeme, M. Nuttin, T. Tuytelaars and L. J. Van Gool, ”Omnidirectional Vision Based Topological Navigation“, In *International Journal of Computer Vision*, 74(3), pp. 219-236, September 2007.

[5] Christoffer Valgren and Achim J. Lilienthal and Tom Duckett, ”Incremental Topological Mapping Using Omnidirectional Vision“, In *International Conference on Intelligent Robots and Systems*, pp. 3441-3447, IEEE, 2006.

[6] Christoffer Valgren and Tom Duckett and Achim J. Lilienthal, ”Incremental Spectral Clustering and Its Application To Topological Mapping“, In *International Conference on Intelligent Robots and Systems*, pp. 4283-4288, IEEE, 2007.

[7] D. G. Lowe, “Distinctive image features from scale-invariant keypoints”, International Journal of Computer Vision, Vol. 60, No 2. pp. 91-110. 2004

[8] Handbyul Joo, Yekeun Jeong, Olivier Duchenne, Seong-Young Ko and In-So Kweon, “Graph-based Robust Shape Matching for Robotic Application”, *in IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009.

[9] Pedram Azad, Tamim Asfour and Radiger Dillmann, “Combining Harris Interest Points and the SIFT Descriptor for Fas Scale-Invariant Object Recognition”, *in IEEE International Conference on Intelligent Robots and Systems*, St. Lois, USA. October 2009.

[10] Ming Liu, Davide Scaramuzza, Cédric Pradalier, Roland Siegwart and Qijun Chen, “Scene recognition with Omnidirectional Vision for Topological Map using Lightweight Adaptive Descriptors”, *in*

IEEE Internatioanal Conference on Intelligent Robots and Systems, St. Lois, USA. October 2009.

[11] Xiaopeng Chen, Qiang Huang, Peng Hu, Min Li, Ye Tian and Chen Li, "Rapid and Precise Object Detection based on Color Histograms and Adaptive Bandwidth Mean Shift", *in IEEE Internatioanal Conference on Intelligent Robots and Systems*, St. Lois, USA. October 2009.

[12] Jianxin Wu, Henrik I. Christensen and James M. Rehg, "Visual Place Categorization: Problem, Dataset, and Algoritm", *in IEEE International Conference on Intelligent Robots and Systems*, St. Lois, USA. October 2009.

[13] R. Vaquez-Martin, R. Marfil and A. Bandera, "Affine image region detection and description", *Journal of Physical Agents*, Vol 4, No. 1. pp. 45-54. 2010.

[14] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Van Gool, A comparison of affine region detectors. In *International Journal of Computer Vision*, 65(1/2):43-72, 2005.

[15] J.Matas, O. Chum, M. Urban, and T. Pajdla, Robust wide baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*, pp. 384-393, 2002.

[16] K. Mikolajczyk and C. Schmid, Scale and Affine invariant interest point detectors. In *International Journal of Computer Vision* 60(1):63-86, 2004.

[17] Yining Deng and B.S. Manjunath, "Unsupervised Segmentation of Color-Texture Regions in Images and Video", *in IEEE Trans. Pattern Anal. Mach. Intell*, vol. 23, No. 8. pp. 800-810. 2001.

[18] Wendy Aguilar, Yann Frauel, Francisco Escolano, M. Elena Martinez-Perez, Arturo Espinosa-Romero and Miguel Angel Lozano, "A robust Graph Transformation Matching for non-rigid registration", *Image Vis. Comput.*, Vol. 27. No. 7. pp 897-910. 2009.

[19] Mike Smith, Ian Baldwin, Winston Churchill, Rohan Paul and Paul Newman, The New College Vision and Laser Data Set, *I. J. Robotic Res*, vol. 28, No. 5, pp. 595-599. 2009.

[20] A. M. Romero and M. Cazorla, "Comparativa de detectores de características visuales y su aplicación al SLAM“, In *X Workshop de Agentes Físicos*, Cáceres, Sep. 2009.

[21] Anna Maria Romero, Miguel Cazorla, Pablo Suau and Francisco Escolano, "Graph-Matching Based Method for scene recognition on Omnidirectional Images“, In *International Conference on Intelligent Robots and Systems*, 2010. *(in revision)*